

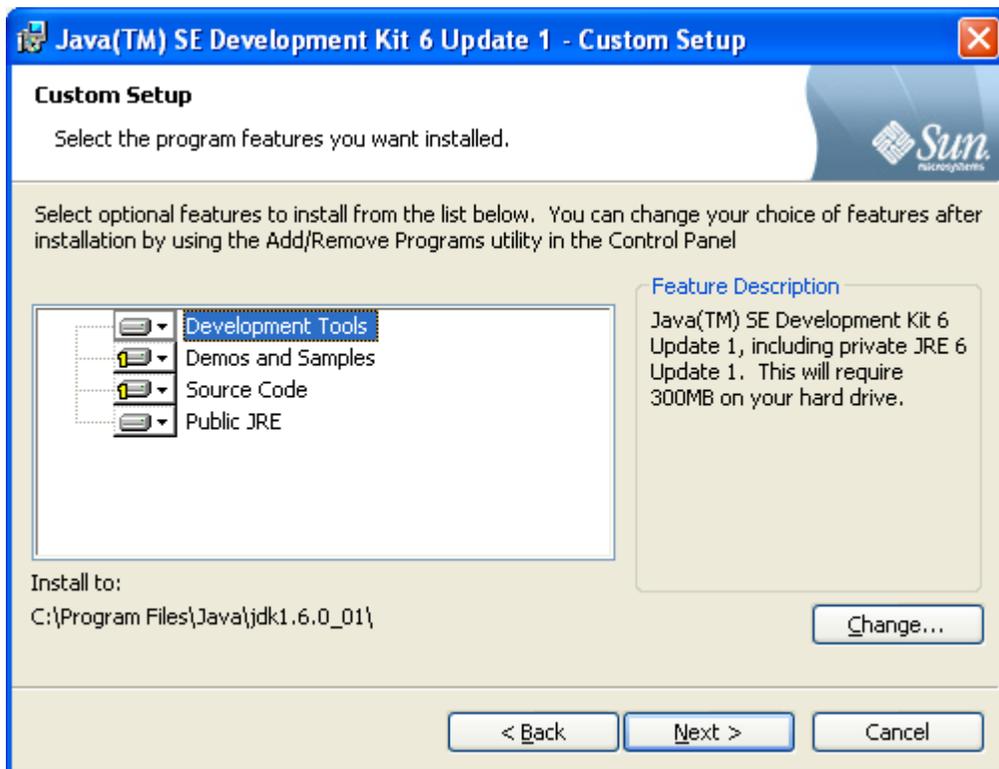
Tutorial 01: Getting Started

Contents:

1. Installing Java.
2. Installing Eclipse.
3. Running Eclipse.
4. Creating and executing a Java application.

Installing Java

1. Download `jdk-6u1-windows-i586-p.exe`.
2. Double-click the file.
3. Accept the default option though you do not need to install 'Demos and Samples' and 'Source Code'.



Installing Eclipse

1. Download `eclipse-SDK-3.2.2-win32.zip`.
2. Right-click the file and choose 'Extract All...'.
3. Extract it to 'Desktop\eclipse-SDK-3.2.2-win32'.



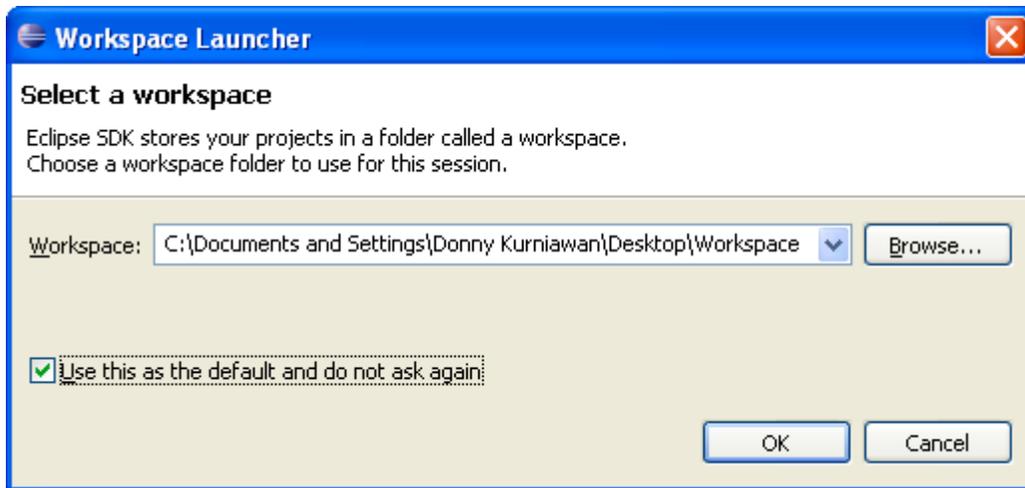
4. Move 'Desktop\eclipse-SDK-3.2.2-win32\eclipse' to 'Desktop\eclipse'.

|| Eclipse does not modify the Windows registry, so it does not matter where it is installed.

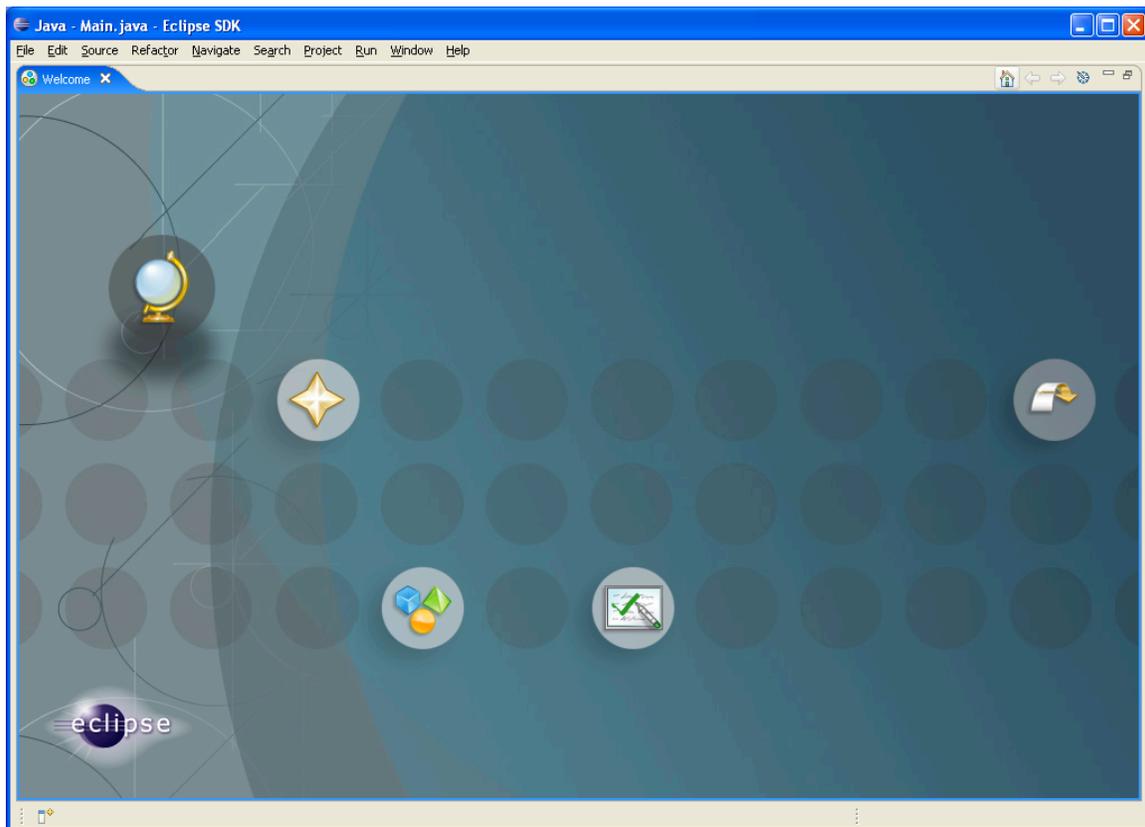
Running Eclipse

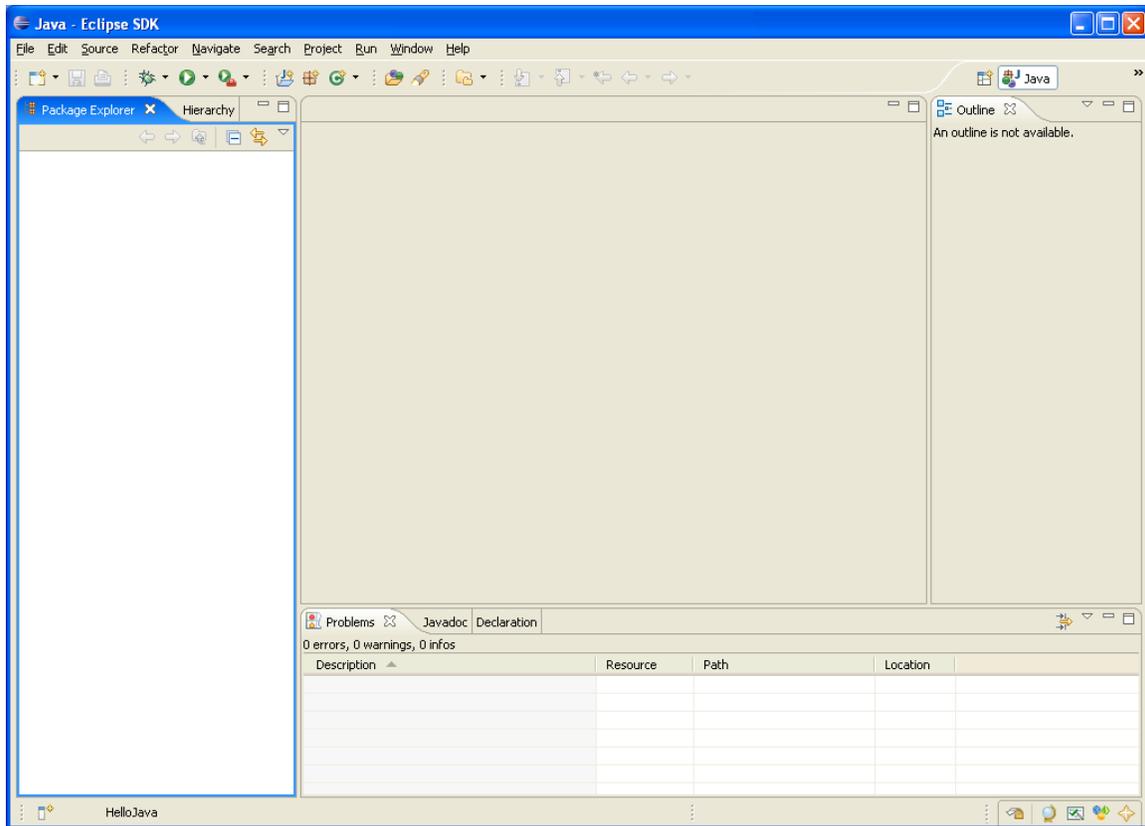
1. Double-click 'Desktop\eclipse\eclipse.exe'.
2. Put 'Desktop\workspace' as the default workspace location.

|| Workspace is a directory hierarchy containing user files such as projects, source code, and plug-in state information such as preferences. One user can have multiple workspaces.



3. Click Workbench.





4. Examine the Eclipse window.

An editor is typically used to edit or browse a document or input object. Modifications made in an editor follow an open-save-close lifecycle model.

A view is typically used to navigate a hierarchy of information, open an editor, or display properties for the active editor.

A collection of various views (Package Explorer, Hierarchy, Outline, Problems, Javadoc, and Declaration) and editors visible within the Eclipse workbench are known as a perspective (e.g. a Java perspective).

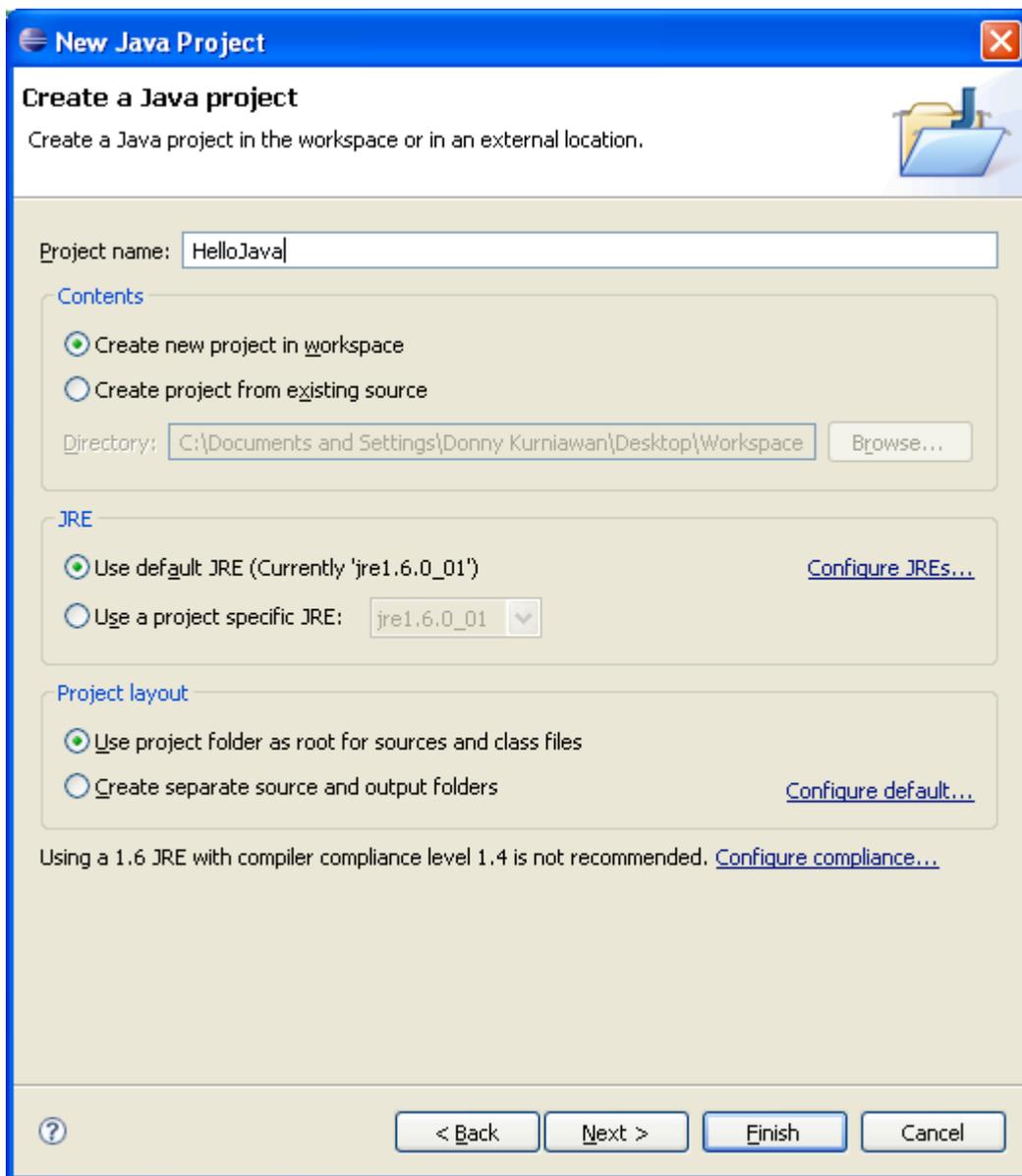
5. Change to a different perspective, click Window > Open Perspective > Other...

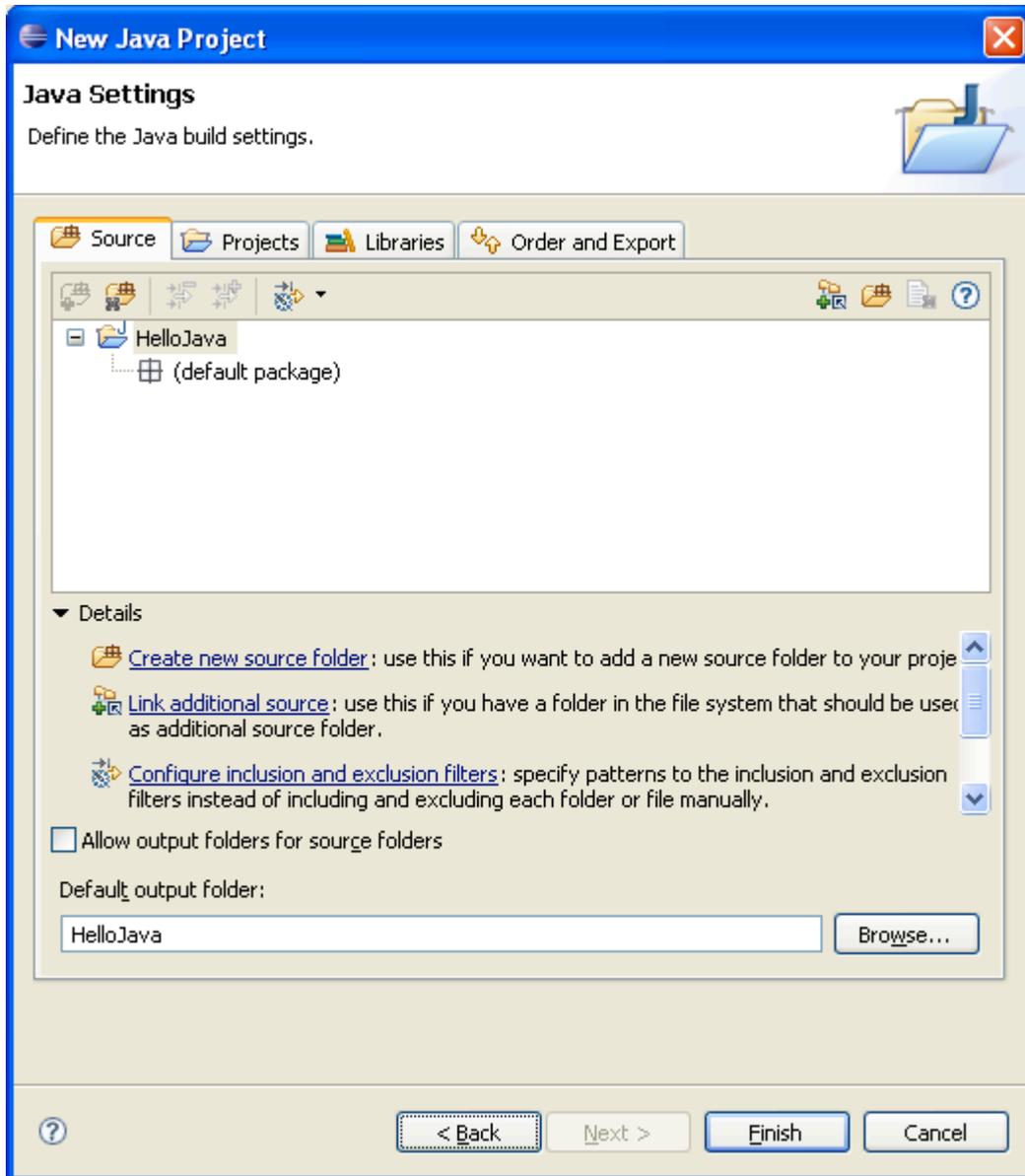
6. Show another view, click Window > Show View > Other...

A view can be moved to another location by dragging its title bar. It can also be stacked on top of another view. It can also be resized, maximized, and minimized.

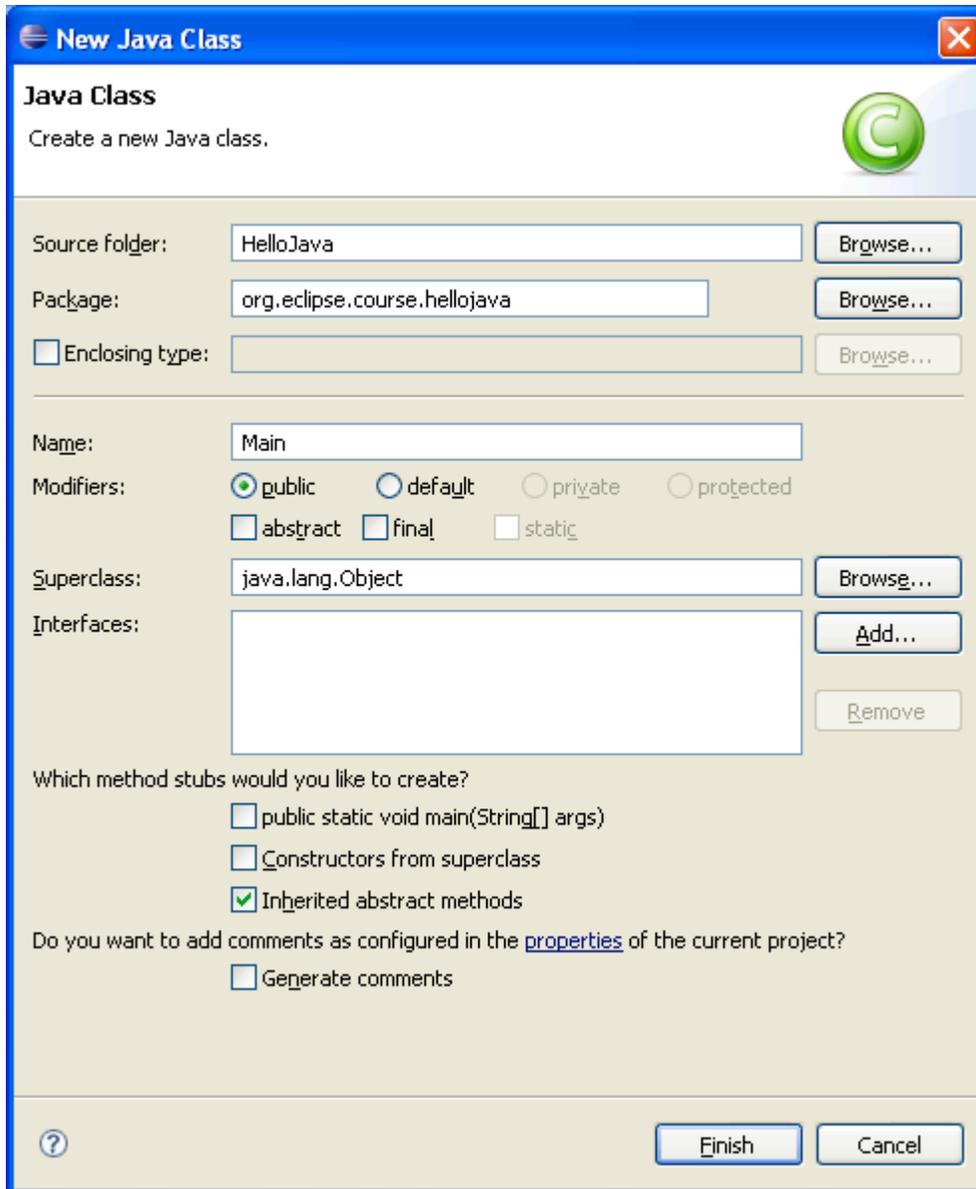
Creating and executing a Java application

1. Change to the Java perspective.
2. Click File > New > Project...
3. Select Java Project.
4. Name the project, HelloJava.





5. Click Finish
6. Switch to the Package Explorer view.
7. Right-click HelloJava, and select New > Package.
8. Put org.eclipse.course.hellojava as the package name.
9. Right-click org.eclipse.course.hellojava package, and select New > Class.
10. Put Main as the class name.



11. Type the following code in Main.java.

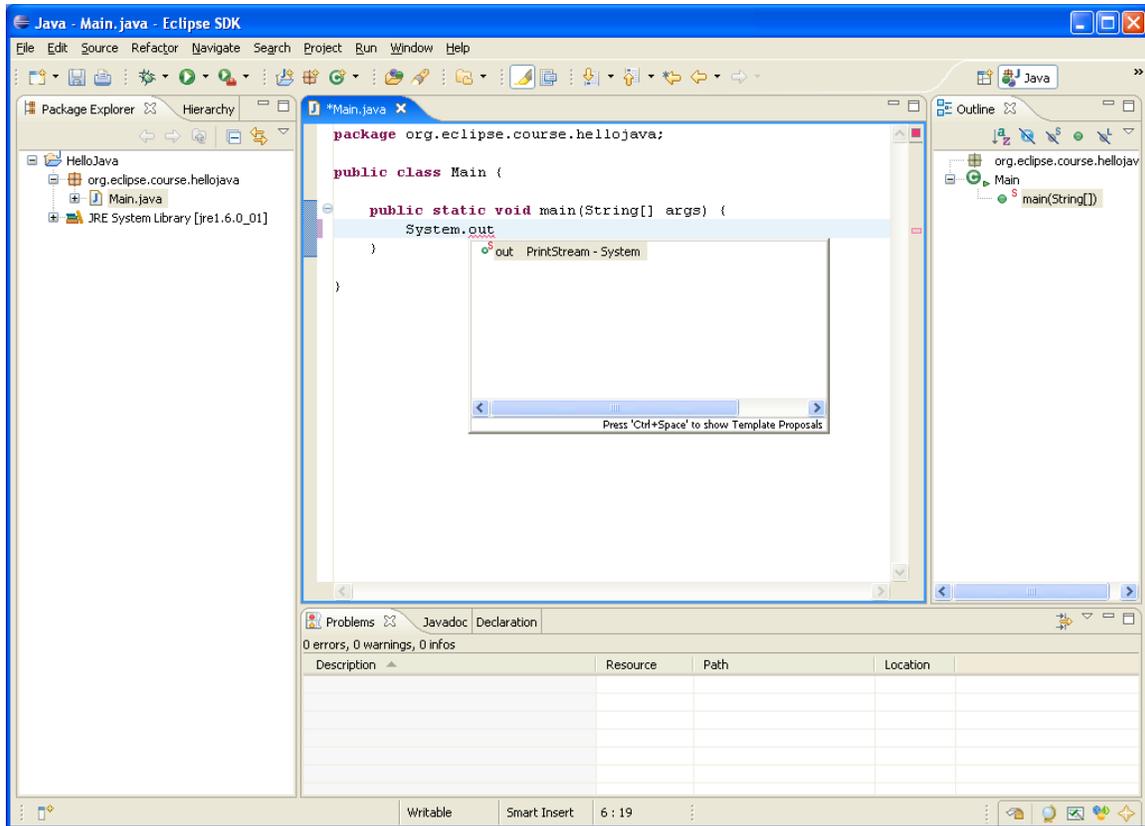
```
package org.eclipse.course.hellojava;

public class Main {

    public static void main(String[] args) {
        System.out.println("Hello Java");
    }

}
```

12. Notice that as you type a period '.', Eclipse offers a list of suggestions (content assist). You can continue to type or you can use the up and down arrows to select the intended keyword.



13. Click Run > Run As > Java Application.

